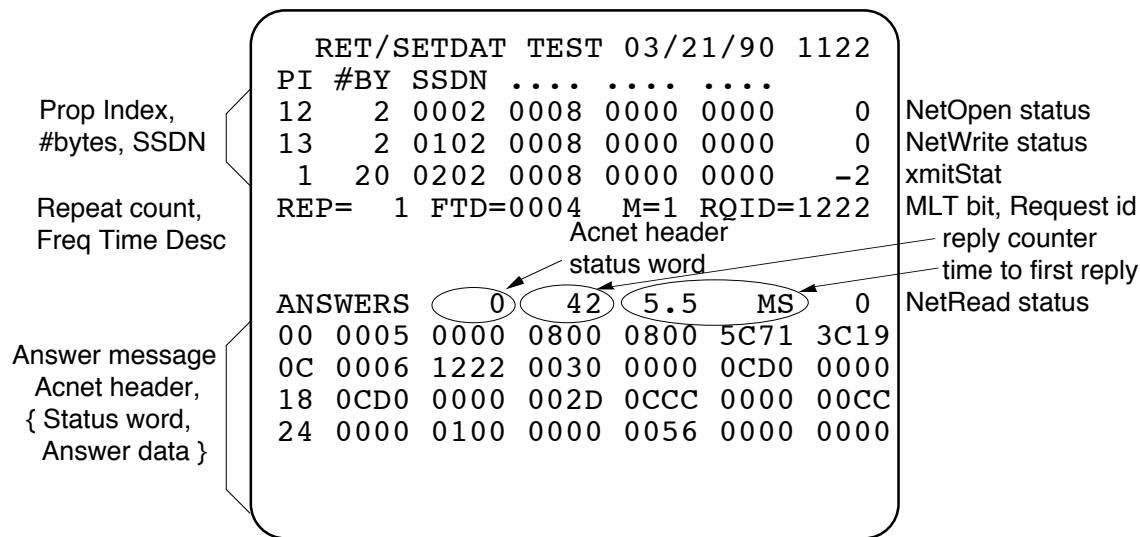# RETDAT-SETDAT Test

Page application
Mar 22, 1990

Introduction

A standard message protocol has been used since 1982 for accelerator control at Fermilab between the console computers and the front end computers. It is referred to locally as "DAS," (Data Acquisition Services) or as "RETDAT/SETDAT," after the network task names that are assigned to receive data requests and settings, respectively.

The Linac control system interfaced to this protocol via the Linac front end PDP–11, which translated between this protocol and the "Classic" protocol that the Local Stations have always supported. The new Linac control system Local Stations will use this standard DAS protocol directly. The VME local stations now support this message format for data requests and for settings. This application page exercises this protocol by issuing data requests or settings to a local station and displaying the responses that are returned. It is written using the local station's Network Layer interface routines.

Display page layout—data request mode


```
        RET/SETDAT TEST 03/21/90 1122
    PI #BY SSDN .... .... ....
    12    2 0002 0008 0000 0000        0      NetOpen status
    13    2 0102 0008 0000 0000        0      NetWrite status
     1   20 0202 0008 0000 0000       -2      xmitStat
    REP=   1 FTD=0004   M=1 RQID=1222         MLT bit, Request id
                        Acnet header              reply counter
                        status word               time to first reply
    ANSWERS        0     42   5.5     MS     0   NetRead status
    00 0005 0000 0800 0800 5C71 3C19
    0C 0006 1222 0030 0000 0CD0 0000
    18 0CD0 0000 002D 0CCC 0000 00CC
    24 0000 0100 0000 0056 0000 0000
```

Labels (left): Prop Index, #bytes, SSDN; Repeat count, Freq Time Desc; Answer message Acnet header, { Status word, Answer data }

This snapshot of the test program display serves as an example of several of its features. It shows an example of requesting a 2-byte reading, a 2-byte setting, and a 20-byte analog alarm block all from channel 0 in node 08. The FTD specification indicates that the data is to be collected at intervals of 15 Hz (66 msec).

Parameter entry for data requests

The M bit is set for periodic requests. The request remains active until a cancel message is sent. If the M bit is zero, the request is automatically cancelled after update of the first set of reply data. The RQID is the request id that serves to uniquely identify one request from another.

Enter up to three device specifications. The Property Index and the #bytes requested/device are specified in decimal. The SSDN (SubSystem Device Number) is entered in hex. It is a 4-word field that is normally extracted from the central database given a property index and a

device index. (The reference that describes these terms more fully is Acnet Design Note 22.28.) To omit a device spec, blank out the property index field.

The Property Index values of interest are:

1	Analog alarm block
3	Basic Control
4	Basic Status
5	Digital alarm block
7	Extended status
12	Reading
13	Setting

The formats for the 4-word SSDN used in Linac are as follows:

listype#	idLng=2
node#	
Chan#/Bit#	
—	

or

listype#	idLng=3
node#	
memory address	

These examples give the long ident forms for channels, bits and addresses. The short form is also supported for backwards compatibility, where the `idLng=1` word for channels and bits, and `idLng=2` words for addresses.

The Frequency Time Descriptor (FTD) expresses the repetition period in 60 Hz cycles. The REP is the repeat count for testing the timing for longer requests. It merely repeats the same set devices to build a request with up to 3*REP request packets, each of which requires 16 bytes in the network data request. Thus, if you enter a count of 100 for a single device spec, the request message is more than 1600 bytes in length.

Interrupt anywhere in the parameter specification area of the screen (on rows 2–8) to initiate the data request. The word ANSWERS in the response area of the screen should be hi-lited to indicate the request is active. (If it is a one-shot request, or if an error is returned, it may not stay hi-lited for long.) When the first (or only) response is received, the elapsed time is displayed on the ANSWERS line in milliseconds. This time is measured from just before the call to `NetWrite` until just after the call to `NetRead` that returns the first response in the test program.

Just to the left of the elapsed time to first response is the decimal count of replies received. The Acnet header status word is also shown in decimal just after the word ANSWERS. This is done for convenience in interpreting error codes, which are negative numbers.

Other status replies that are shown are toward the right end of the screen. The status return from `NetOpen`, which is called upon entry to the page, is given at the end of the third line. Below that is the return from the call to `NetWrite` when the request message was issued. Below that is the transmit status word that gives the success of the network transmission. At the end of the ANSWERS line is the status return from the call to `NetRead` which returns the reply data.

Cancelling data requests

Cancel an active request by an interrupt on the ANSWERS line. The hi-liting will be removed as the cancel message is sent (a USM with bit#9 set in the Acnet header flags word). (Note that an interrupt on this line when a request is not active switches to the setting mode.) Leaving the page results in all active requests being cancelled the next time a reply is received. This happens because the application closes its network connection (via NetClose) upon exit.

Answer data viewing

Six lines are used to display answer data in hex with 6 words per line. The byte value at the left shows the offset in bytes of the first word on the line. There are three ways to adjust the starting offset for the block of answers.

The easiest way is to use the raise/lower buttons on the local console. It will adjust the offset by 72 bytes (6 lines of 6 words each) at a time. If you advance too far, it wraps to the start.

To adjust the offset so that the first word is one of the displayed data words, merely interrupt under the word you want to be the starting word displayed. Obviously, you can only move forward in this way.

The third way to adjust the offset is by typing in the desired offset in the first characters of the first line of answer data and interrupting. Three characters can be entered, even though only the least significant two characters can be displayed due to the screen's limited number of characters per line.

The entire response message is displayed, beginning with the 9-word Acnet header. (For an error response, this is all you get.) This is followed by a reply packet for each device, consisting of an error status return code followed by the answer data for that device, padded to an even number of bytes.

Details of the example response

The first 9 words are the Acnet header. The first word shows that it is a reply message with the MLT bit set. The next word is the reply status word, and it is also shown in decimal on the line above. The destination and source bode of the request are both node 08 in this case. This example illustrates use of the network to make a request to itself. (If it didn't do that, it would always require two nodes to do the test.) A by-product of this is that the xmitStat value shown at the end of the fifth line is -2, indicating "address not recognized," which in this case is normal.

The next two words of the Acnet header are the destination task name, which for Acnet data requests is RETDAT, in the Radix-50 encoding commonly used by PDP-11 computers. The source task id is 6, which denotes the table index in the Network Connect Table returned by the call to NetOpen. The request id is followed by the message size word, which is the total size of the response message including the Acnet header itself.
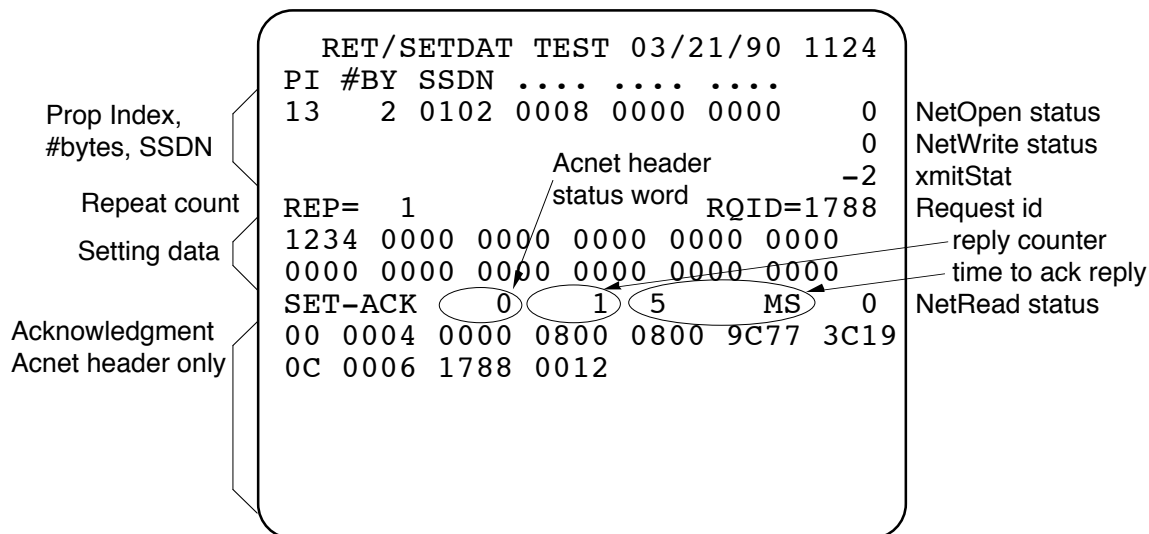
The Acnet header is followed by the reply packets. In each case, the error status word is zero, indicating no errors. Both the reading and setting of the device are 0CD0. The analog alarm block includes the nominal and tolerance values of 0CCC and 00CC, respectively. The value 0056 is the trip counter.

The time response value for this example is 5.5 msec. This seems to be a common value for a one-shot minimal data request in any of the three protocols supported by the Local Station software. As of the time the snapshot was taken, there had been 42 replies received.

Switch between request/setting modes

Interrupt on the ANSWERS line to switch into setting mode from request mode. Interrupt on the SET-ACK line (same one) to switch back to request mode from setting mode.

Display page layout—setting mode

```
        RET/SETDAT TEST 03/21/90 1124
     PI #BY SSDN .... .... ....
     13   2 0102 0008 0000 0000      0     NetOpen status
                                     0     NetWrite status
                  Acnet header      -2     xmitStat
     REP=   1    status word  RQID=1788    Request id
     1234 0000 0000 0000 0000 0000          reply counter
     0000 0000 0000 0000 0000 0000          time to ack reply
     SET-ACK      0      1  5      MS   0   NetRead status
     00 0004 0000 0800 0800 9C77 3C19
     0C 0006 1788 0012
```

Prop Index, #bytes, SSDN

Repeat count

Setting data

Acknowledgment Acnet header only

Parameter entry for settings

The device spec data is entered the same way as for requests. In addition, the setting data may be entered. There are 24 bytes of setting data available. If more setting data is required to satisfy the device specs, the setting data words are simply repeated. When the REP (repeat count) is more than one, the setting data is re-used starting at the beginning for each repetition. No FTD value nor M bit value is used for settings.

Setting acknowledgment

The reply to a setting is the status-only acknowledgment message. It consists only of the Acnet header with the status code in the second word. For the example shown, The 0004 indicates a reply messages, the destination node and source nodes are both node 08 as before, the destination task name was SETDAT in Radix-50 encoding, the source task id was 6, the request id was 1788, and the total length of the reply message is 18 bytes. The time to respond was 5 msec.